AMENDMENTS TO THE CLAIMS

This listing of claims replaces all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (Currently Amended) In a computing environment, a method comprising:

receiving an original model comprising a plurality of elements representing operations to test;

receiving as input an interger N representing a length of subsequences, N being greater than or equal to 2;

receiving as input an integer M representing the maximum length of a test case;

generating ~~producing~~ a list of fixed length subsequences from the elements, each subsequence comprising ~~a predetermined and fixed number of~~ N elements, ~~the number being at least two,~~ and the list comprising all subsequences such that each subsequence of elements of length N ~~equal to the fixed number~~ is represented within the list;

removing from the list of subsequences each subsequence having been designated as invalid; and

generating a ~~suite~~ list of test cases, each test case comprising M elements, from the list of subsequences, such that each valid subsequence appears at least once within the list of test cases. ~~among the test cases in the suite.~~

2. (Original) The method of claim 1 further comprising, evaluating a constraint against the test cases to determine for each test case whether that test case matches the constraint.

3. (Original) The method of claim 2 wherein the constraint matches a selected test case, and further comprising, splitting the test case into at least two test cases.

4. (Original) The method of claim 1 further comprising, evaluating a precondition against the test cases to determine for each test case whether that test case matches the precondition.

5. (Previously Presented) The method of claim 4 wherein the precondition matches a selected test case, a further comprising, splitting the test case into at least two test cases.

6. (Original) The method of claim 1 wherein a plurality of preconditions are known, and further comprising, sorting the preconditions into an order, and evaluating each precondition against the test cases based on the order to determine for each test case whether that test case matches the precondition.

7. (Currently Amended) The method of claim 1 wherein generating a ~~suite~~ list of test cases from the subsequences comprises selecting a subsequence based on a selection algorithm, and adding the selected subsequence to a test case.

8. (Original) The method of claim 7 further comprising, marking the selected subsequence as covered.

9. (Original) The method of claim 8 further comprising, selecting another subsequence from a set of uncovered subsequences, and adding the other subsequence to the test case.

10. (Original) The method of claim 8 wherein selecting a subsequence based on a selection algorithm comprises, determining which element starts a largest number of still uncovered subsequences, and selecting a subsequence starting with that element.

11. (Original) The method of claim 8 wherein selecting a subsequence based on a selection algorithm comprises, determining which element starts a largest number of still uncovered subsequences, and if there is only one such an element, selecting a subsequence starting with that element, and if there is a tie, employing a tiebreaker

12. (Original) The method of claim 1 wherein generating a suite of test cases from the subsequences comprises selecting a subsequence, adding the selected subsequence to a test case, marking the selected subsequence as covered, and repeating until no subsequence remains uncovered.

13. (Previously Presented) A computer-readable storage medium having computer-executable instructions for performing the method of claim 1.

14. (Currently Amended) A computing system comprising one or more computer processors and computer readable storage media and upon which is executing the method of claim 1.

~~In a computing environment, a system comprising:~~

~~means for receiving a model file comprising a plurality of elements representing operations to test;~~

~~means for producing list of subsequences from the elements, each subsequence comprising a predetermined and fixed number of elements, the number being at least two, and the list comprising all subsequences such that each subsequence of elements of length equal to the fixed number is represented within the list; ;~~

~~a means for removing invalid subsequences from the list; and~~

~~means for generating a suite of test cases from the list of subsequences, including means for selecting a subsequence, means for adding the selected subsequence to a test case, means for marking the selected subsequence as covered, such that any valid subsequence is covered at least once among the test cases in the suite.~~

15. (Original) The system of claim 14 further comprising, evaluation means for evaluating a constraint against the test cases to determine for each test case whether that test case matches the constraint, and if so, the evaluation means employing means for splitting that test case into at least two test cases.

16. (Previously Presented) The system of claim 14 further comprising, evaluation means for evaluating a precondition against the test cases to determine for each test case whether that test case matches the precondition, and if so, the evaluation means employing means for splitting that test case into at least two test cases.

17. (Currently Amended) The system of claims 14 wherein the method executing further comprises ~~means for selecting a subsequence comprises means for counting which~~ determining an element which starts a still most uncovered number of subsequences.

18. (Currently Amended) A computer program product comprising physical computer readable storage media upon which is encoded instructions which, when executed, perform the method of claim 1.

~~In a computing environment, a system for constructing test cases, the system comprising:~~

~~a model file reader that receives a model file comprising elements, each element representing an operation to test;~~

~~a pre-processing mechanism that produces a list of subsequences, each subsequence in the list of the same fixed length and comprising at least two elements, and the list of subsequences comprising each combination of elements of fixed length from the model file;~~

~~a selection mechanism that selects a subsequence from the list of subsequences, adds the subsequence to a test case, and marks the subsequence as covered; and~~

~~the selection mechanism controlled to add subsequences to test cases until all subsequences within the list have been marked as covered.~~

19. (Currently Amended) The computer program product ~~system~~ of claim 18 wherein the method further comprises evaluating a constraint against the test cases to determine for each test case whether that test case matches the constraint. ~~further comprising a post-processing mechanism that is configured to split a single test case into at least two test cases based on any preconditions or constraints associated with the model file.~~

20. (Currently Amended) The computer program product ~~system~~ of claim 18 wherein the method further comprises evaluating a precondition against the test cases to determine for each test case whether that test case matches the precondition. ~~the selection mechanism employs a selection algorithm to select a subsequence, by determining which element starts a largest number of still uncovered subsequences, and if there is only one such an element, selecting a subsequence starting with that element, and if there is a tie, employing a tiebreaker.~~

21. (Currently Amended) The computer program product ~~system~~ of claim ~~20~~ 18 wherein the method further comprises sorting a set of preconditions into an order, and evaluating

each precondition against the test cases based on the order to determine for each test case whether that test case matches the precondition. ~~tiebreaker comprises selecting a first available subsequence.~~

22. (Currently Amended) The computer program product ~~system~~ of claim ~~20~~18 wherein, within the method, generating the list of test cases from the subsequences is based up selecting a subsequence based on a selection algorithm, and adding the selected subsequence to a test case. ~~selection mechanism adds another subsequence to the test set based on a subsequence already in the test set.~~

23. (Currently Amended) The computer program product ~~system~~ of claim 22 wherein, within the method, selecting a subsequence based on a selection algorithm comprises determining which element starts a largest number of still uncovered subsequences and selecting a subsequence starting with that element. ~~the selection mechanism first looks for the other subsequence among uncovered subsequences in the set, and if non can be used, secondarily looks for a covered subsequence in the set.~~